



자료구조 기초

너만보여 스터디

학습 내용

1. 중요 문법 리뷰
2. 숙제 리뷰
3. 차기 스터디 공지



보강내용

기본적인 자료구조

- 배열(Array) : 데이터를 순차적으로 저장해 0부터 시작하는 인덱스를 통해 접근한다.
 - . 일반적으로 배열은 선언할 때 크기 고정
 - . 데이터를 임의 접근할 수 있어 접근을 효율적으로 할 수 있음
- 리스트(List) : 배열과 유사한 순차적인 자료구조를 제공하며 데이터 접근을 위해 인덱스를 사용해야 하는 점은 배열과 같지만 배열과 달리 초기에 크기를 고정하지 않는다.
 - . 데이터 크기가 고정되지 않음
 - . 데이터를 다루기 위한 여러 방법이 제공됨
 - . 리스트의 데이터는 서로 다른 타입일 수 있음
 - . 배열 중간에 값을 추가하거나 삭제하기 쉬움

→ ArrayList
- 맵(Map) : 데이터를 Key-Value(키-값)의 쌍으로 저장하는 방식이다.
 - . 맵을 사용했을 때 얻을 수 있는 가장 큰 장점은 원하는 데이터를 빠르게 찾을 수 있다는 점이다.
 - . 데이터를 저장할 때 해당 데이터를 찾기 위한 Key를 부여
 - . Key값을 알면 언제든지 빠르게 데이터를 찾을 수 있음
 - . Value 에 객체형이 들어갈 수 있어 복잡한 데이터 처리가 가능

→ HashMap, Hashtable



보강내용

함수와 데이터

- 기본적으로 함수는 독립적 모듈
 - . 호출하는 함수에서 호출되는 함수로 데이터를 전달 할 때는 인자(parameter) 이용
 - . 호출되는 함수에서 호출하는 함수로 데이터를 전달 할 때에는 리턴값을 이용
- 함수안에 변수나 자료구조 배치 → 함수내에서 선언
 - . 임시적으로 저장해야 하는 데이터
- 함수 밖에 변수나 자료구조 배치 → 멤버변수, 필드
 - . 멤버함수(메소드) 사이에서 공유해야 할 때
- 전체에서 공유하는 자료구조나 변수 → static 멤버변수
 - . 프로그램 전체를 통해 공유해야 할 때



공지 사항

공통 과제

1. 숫자 야구 게임 만들기 → 소스 변형

1. Game Class

: 게임 메인 프로세스

: 답 입력 받기 함수

: 문제, 답 비교 함수

2. MakeNumber Class : 문제 만들기 클래스

```
MakeNumber maker = new MakeNumber();
```

```
int[] comNumber = maker.makeNum();
```

3. PrintResult Class : 결과 출력 클래스

```
PrintResult printer = new PrintResult();
```

```
printer.print(userNumber, resultText);
```

```
* String[] resultText = new String[gameCount];
```

```
Integer[][] userNumber = new Integer[gameCount][3];
```

